

Education needs Free/Libre Software needs Education

Walter Bender and Devin Ulibarri

Copyright 2016 CC-BY-SA



Talk outline

- (1) A brief argument for Constructionist learning;
- (2) An argument for Free/Libre Software for Constructionist learning;
- (3) Concrete steps towards integration of learning with Free/Libre Software;
- (4) Discussion: How can we advocate for Free/Libre Software in Education?

Testimonial

Devin Ulibarri, Classical Guitar Teacher



“Education either functions as an instrument which is used to facilitate integration of the younger generation into the logic of the present system and bring about conformity or it becomes the practice of freedom, the means by which men and women deal critically and creatively with reality and discover how to participate in the transformation of their world.”

—Paulo Freire, *Pedagogy of the Oppressed*



A simple goal

To give every child an opportunity for a quality and equitable education.



A (misguided) focus on efficiency

Learning is not a service or an App or a video.

Learning is not a “flipped classroom” or a tablet.

Learning is not curriculum.

Learning is not efficient.



“[We are developing the capability] to do in nanoseconds things that we shouldn’t be doing at all.” —Gerald Bracey



Proprietary software in education is, at best, a *missed opportunity* to:

- learn new skills;
- exercise social etiquette;
- build upon preexisting ideas; and
- embrace personal responsibility.

Proprietary software in education is an *injustice* because:

- developers have unjust control over users;
- capitalization of the unbalanced power dynamic.

Intersection of Free/Libre Software and education policy

- Examination of common arguments.
- How does software licensing affect education?
- What choices are being made everyday that impact education?
- What are things we can do as stakeholders to guide the future of education policy?

But not every child will become a...

- The usual argument is, “well not every child will become a [fill in the blank], so why should we give them [x,y,z]?”!



Human Potential and Social Justice

- Education Policy (and the realization of such policy) should begin with the assumption that every human being has unlimited potential.

Affects on learner mindset

Ulibarri Hypothesis (part 1/2):

Children presented with a *proprietary software paradigm* will be much less motivated to learn.

Affects on learner mindset

Ulibarri Hypothesis (part 2/2):

Children presented with a *software freedom paradigm* will be *highly motivated to achieve and learn much more*.

Freedom and potential?

In the case of software, it is the *freedoms afforded users that allow its users to exercise their potential*. We wouldn't want any less from teaching/learning in any subject.

What if there were a Free/Libre Software policy for schools?



Libre EDU (1 of 4)

Children can utilize Free/Libre Software tools in all of their classes.

Libre EDU (2 of 4)

Computer classes can benefit now from
studying source code of familiar software.



Libre EDU (3 of 4)

Technology can be taught as having *social* and *moral implications*.



Libre EDU (4 of 4)

Students feel part of a *dynamic community*—not “just users”.



Implications? Wait, there is more!!

“The context of human development is always a culture, not an isolated technology.”

—Seymour Papert

Learning is a verb.

Free/Libre Software puts an emphasis on *powerful tools*.

It is a *culture of doing* and sharing in a context of critique, reflection, and personal responsibility.

Why would we ever want to draw a hard line between tool makers and tool users?



The Maker Movement and Free/Libre Software Kindred spirits (almost)

- Individual creativity at their cores; but...
- Neither has of yet penetrated the classroom;
and
- The Maker Movement is being co-opted.

@LibrePlanet 2010

“Only Free/Libre Software had achieved the elusive goal of being *write once, run everywhere*.”
—Eben Moglen



@EDU

In the context of learning, we want our code to be *written over and over again* by our end users because they will *learn* in the process.

“Debugging is one of the most powerful educational ideas of the 21st century.”

—Cynthia Solomon

Metrics

The usual metrics—robustness, efficiency, maintainability, etc.—are not enough for education.

If more code is written with end-user contributions in mind, we will have a world in which everyone is engaged in debugging.

Scaffolding 1 of 3

- Establishing a culture in which it is the norm to exercise freedom;
- Provide easy to access the source;
- Use scripting languages so that changes can be direct and immediate;

Scaffolding 2 of 3

- Provide a scaffolding to enable the end user to get started by taking small steps;
- If the penalty for a mistake is too high then people will quickly be conditioned not to engage in the “risky” behavior of modifying code;
- Make learning “visible”;

Scaffolding 3 of 3

- If you can only modify toy versions of your software, you cannot scratch a real itch;
- Be welcoming and tolerant of “newbies”;
- Be unyielding on personal privacy—these are children, after all.

Where are the patches?

In 2010, John Gilmore asked how many patches have been contributed by Sugar users?

In 2015, more than 50% of the patches came from children.

Sugar Labs' lead developer is 15-years old.

The pundits speak out.

“Teachers should not have to learn GIT.”
—Gary Stager

“Teachers can learn too!” —Walter Bender



“Our mission is to promote student achievement and preparation for global competitiveness by fostering educational excellence and ensuring equal access.”

United States Department of Education
<http://www.ed.gov/>

Equity and excellence

Free/Libre Software is *equitable* because...

- the distribution of power is equitable; and
- it can be shared, redistributed.

Equity and excellence

Free/Libre Software is *excellent* because...

- users are in full ownership;
- freedom is valuable; and
- it is only feasible way to ensure user privacy.

Let's be fair! Let's be equitable!

Non-free software is *not fair*...

- ...to *students* because they can not, may not share/distribute software to their peers;
- ...to *families* because it gives developers/companies the license, the power to invade privacy of their children;
- ...to *teachers* because they are left to beg for pedagogical features that they really need in the classroom;
- ...to *everybody* because it gives software companies such an unfair market advantage.

Let's be fair! Let's be equitable!

Free/Libre Software is *fair* and *equitable*...

- ...to *students* because they can share/distribute their software tools (Let's encourage and help them to do this!);
- ...to *families* because they are empowered to audit any possible privacy issues scripted into the software;
- ...to *teachers* because it empowers them to contribute to the development of features they actually need in the classroom;
- ...to *everybody* because it gives everyone a level playing field in the free market.

Advocacy

We need to reach out to decision makers and thought leaders...

Free/Libre Software needs to be written into public education policy (just as it is beginning to be written into administrative policy).

Next steps

Let's target three pundits:

Gary Stager

Valerie Strauss

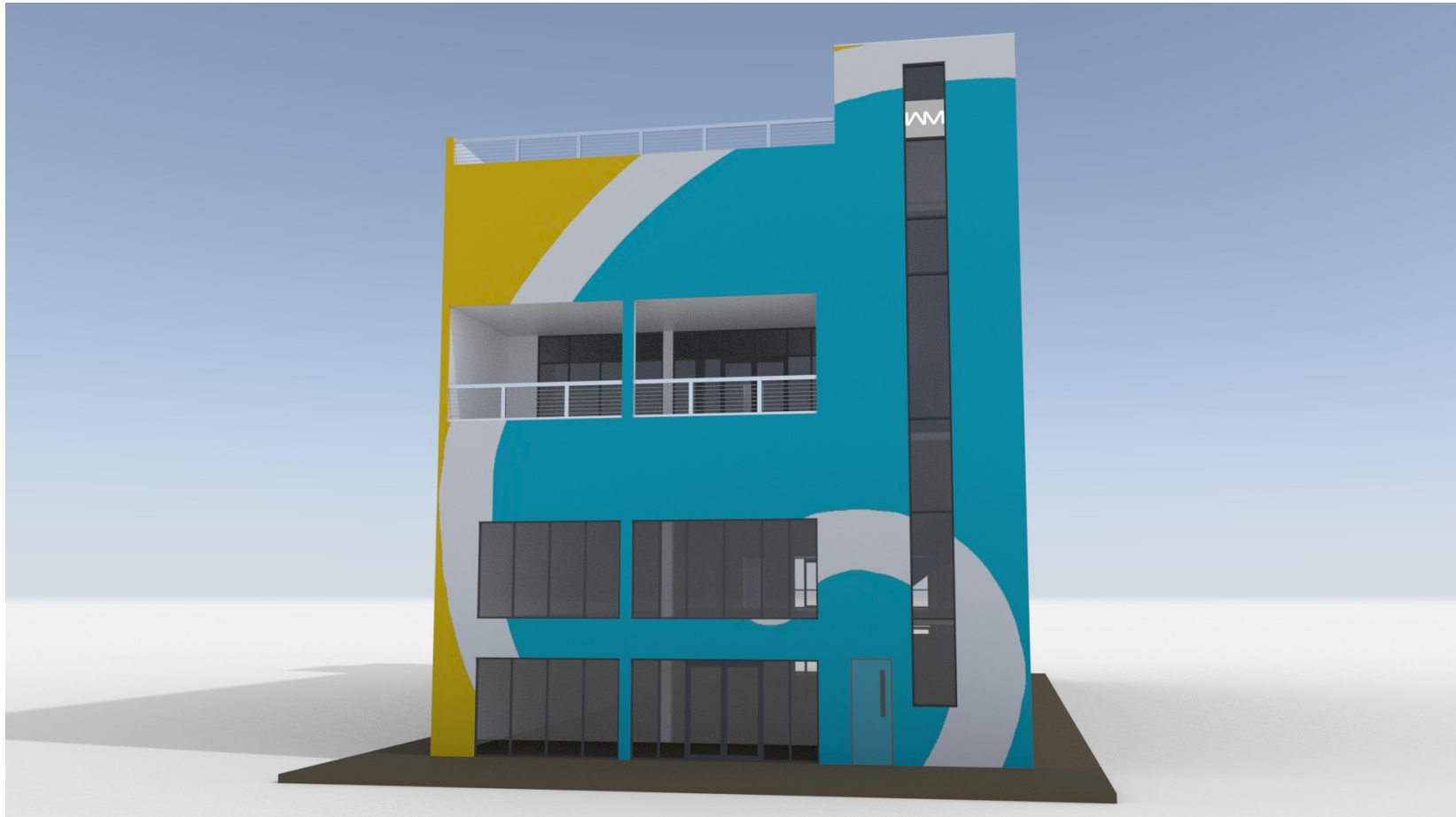
Russell Shilling

Our seed corn

The next generation of computer users are in school today.

We need to ensure that they take advantage of the opportunities afforded by Free/Libre Software.

Coming in 2017: A Free/Libre Software college



Thank you

- Libre Boston Team (www.libreboston.org)
- Encouragement and Guidance from Richard Stallman
- John Sullivan and FSF's hard work and efforts
- Sugar Labs team, which includes the kids!
- Software Freedom Conservancy



Author contact information

devin@devinulibbarri.com

walter@mit.edu





Photo by ??? CC-BY-??